

GETTING STARTED

WAREFAB KONNECT

ESP32 SIGFOX

DEVELOPMENT BOARD

Installing Arduino IDE

Konnnect ESP32 Sigfox dev board can be programmed using either python, Lua, Js, C and C++.

This tutorial will focus on C/C++ using Arduino as the IDE.

Download Arduino IDE

- Go to <https://www.arduino.cc/en/Main/Software> and download the latest release if you don't have the IDE already installed.

Download the Arduino IDE



ARDUINO 1.8.11

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for installation instructions.

Windows Installer, for Windows XP and up
Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10
[Get](#)

Mac OS X 10.8 Mountain Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

[Release Notes](#)
[Source Code](#)
[Checksums \(sha512\)](#)

- Complete the installation instructions and open the IDE.

Installing ESP32 Dependencies

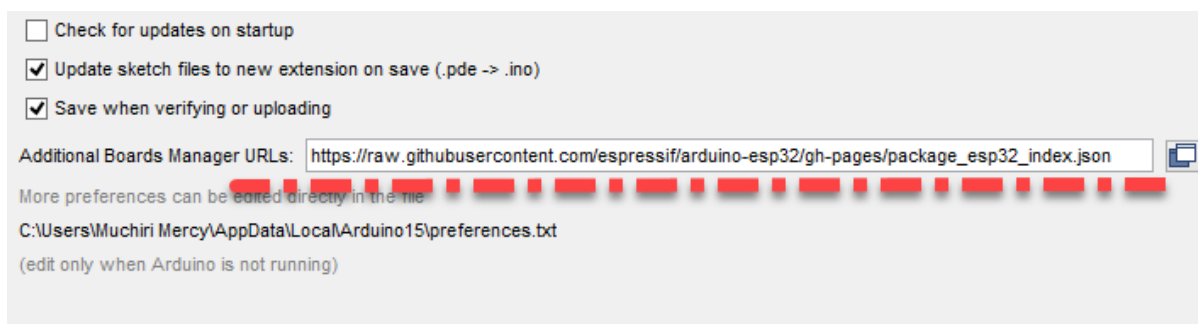
- Go to https://github.com/espressif/arduino-esp32/blob/master/docs/arduino-ide/boards_manager.md and copy the installation board url (stable release).
- Or copy https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

Installation instructions using Arduino IDE Boards Manager

- =====
- Stable release link: https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
 - Development release link: https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_dev_index.json

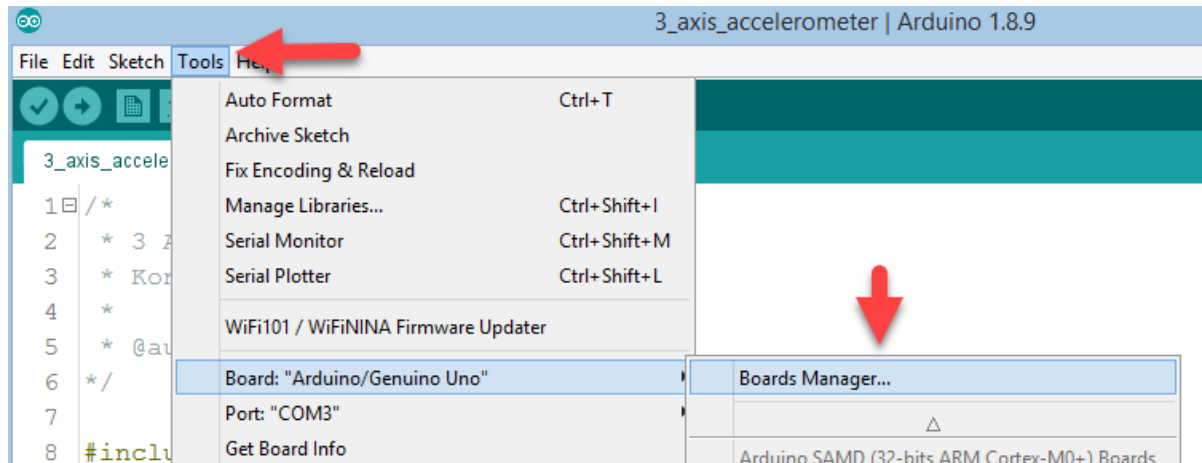
Starting with 1.6.4, Arduino allows installation of third-party platform packages using Boards Manager. We have packages available for Windows, Mac OS, and Linux (32, 64 bit and ARM).

- On the Arduino IDE, open **File-Preferences** and paste the link in the “**Additional Board Manager URLs**”

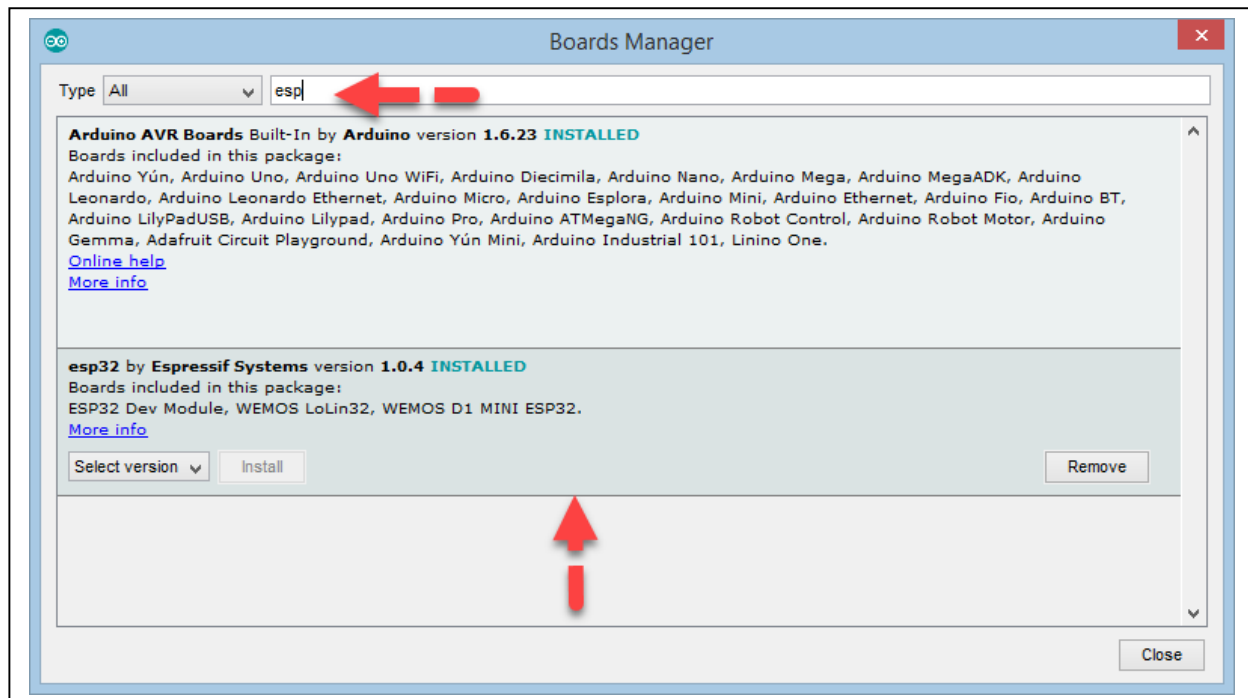


- Close the preferences window

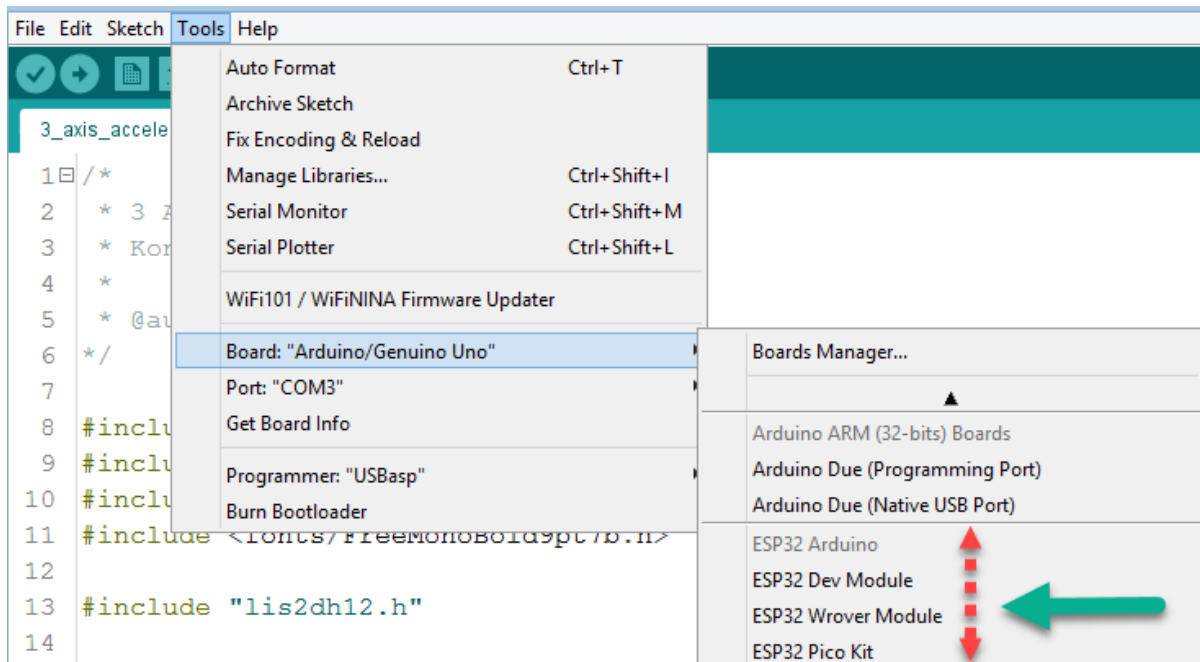
- On the IDE, go to **Tools – Board – Board Manager** to open board manager window.



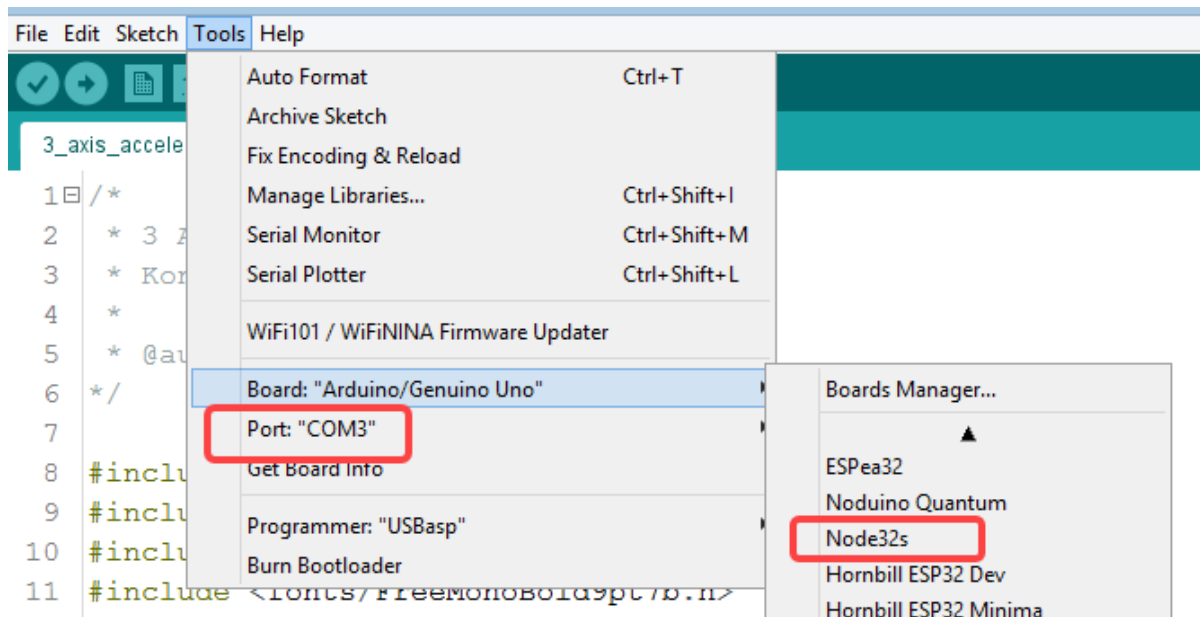
- Search "esp" and install the dependencies



- Confirm if the installation is successful, go to **Tools-Board** and check if there are additional boards for esp32, "ESP32 Arduino", installed.



- Scroll down and select "Node32s"
- Select the **COM Port** the kit is connected to.



Using Konnect ESP32 Sigfox Kit

- Download libraries and examples at <https://github.com/warefab/Konnect-ESP32-Sigfox> and Clone/Download the repo.

warefab / Konnect-ESP32-Sigfox

Unwatch 2 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Warefab Konnect ESP32 Sigfox Dev Board Edit

Manage topics

14 commits 1 branch 0 packages 0 releases 1 contributor BSD-3-Clause

Branch: master New pull request

Create new file Upload files Find file Clone or download

Clone with HTTPS Use SSH

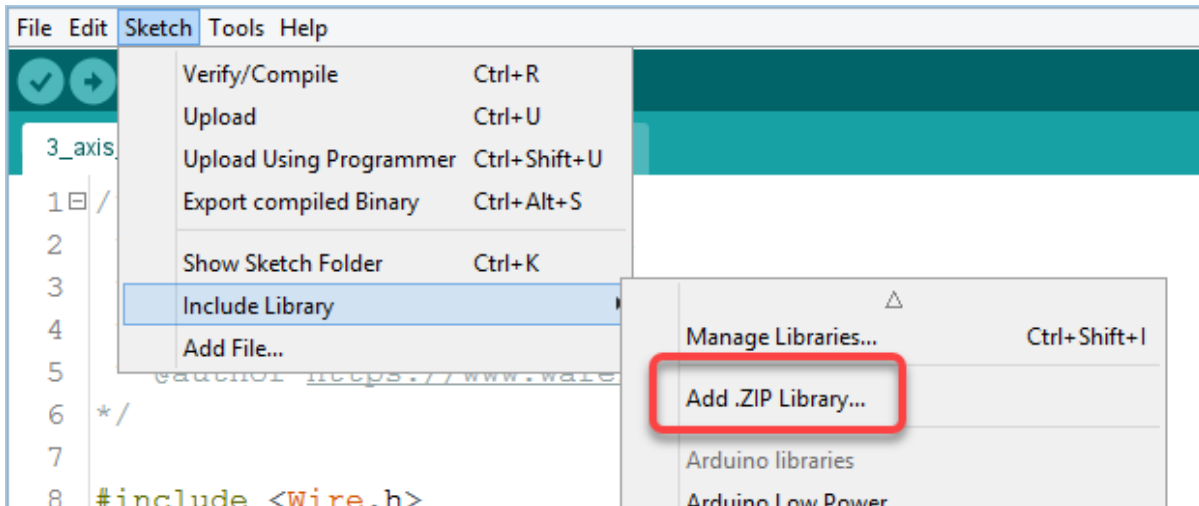
Use Git or checkout with SVN using the web URL.

https://github.com/warefab/Konnect-ESP

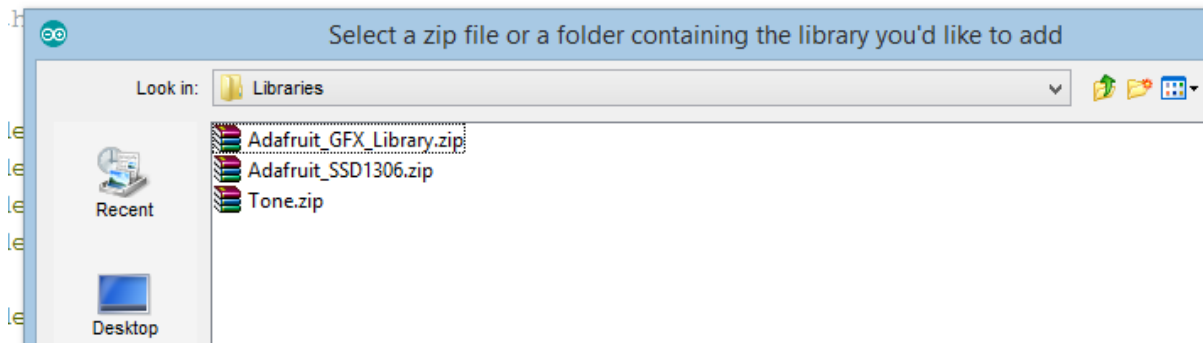
Open in Desktop Download ZIP

muchirjohn sigfox ready example		
3_axis_accelerometer	Examples	
Libraries	Libraries	
Oled_Bitmap	Examples	
analog_mems_mic	Examples	
buzzer	Examples	4 hours ago
gps	Examples	4 hours ago
inbuilt_led	Examples	4 hours ago
light_sensor	Examples	4 hours ago
sht30_t_h	Examples	4 hours ago
sigfox_ready	sigfox ready example	15 minutes ago
sigfox_test	Examples	4 hours ago
.gitattributes	Initial commit	yesterday
LICENSE	Initial commit	yesterday
README.md	Initial commit	yesterday

- To add libraries, on the IDE, go to **Sketch – Include Library – add .ZIP library** to open libraries window



- Navigate to downloaded repo folder, go to libraries and add the zip files.



- Add the Libraries
 - Adafruit_GFX_Library.zip
 - Adafruit_SSD1306.zip
 - Tone.zip

- Open the examples in the repo by clicking the [\[example\].ino](#) file

3_axis_accelerometer.ino	Test 3 axis accelerometer sensor
analog_mems_mic.ino	Test analog mems mic sensor
Buzzer.ino	Test buzzer – creates tones
gps.ino	Test GPS module – gets location
light_sensor.ino	Test ambient light sensor
sht30_t_h.ino	Test temperature and humidity sensor
sigfox_test.ino	Simple Sigfox test
sigfox_ready.ino	Sends kits sensor data to Sigfox cloud

Sending Data to Sigfox Cloud

- Open "Sigfox_ready" example, compile and upload to your kit

Basic Commands

AT\$I=0	Module Version
AT\$I=10	Module ID
AT\$I=11	Module PAC
AT\$SF=	Send Packet to Sigfox cloud

- Once you have have uploaded the example successfully, open serial monitor with baud rate 115200 and reset the kit, press the reset button.
- The module ID and PAC will be displayed in the serial monitor. We'll use this credentials to create new device in the next Sigfox session.

Packet Format

Packet Sample: 01c9a838321e2e01091b3046 [12 Bytes]			
3 Bytes	0x01c9a8	117160	Latitude
3 Bytes	0x38321e	3682846	Longitude
1 Byte	0x2e	46	Light Intensity, %
1 Byte	0x01	1	Acc X axis, g
1 Byte	0x09	9	Acc Y axis, g
1 Byte	0x1b	27	Temperature, Celsius
1 Byte	0x30	48	Humidity, %RH
1 Byte	0x46	01000110	Flags

- Flags - Sound, Light, Acc-X-Neg, Acc-Y-Neg, GPS-Standby, GPS-Speed, GPS-South, GPS-West

Combining the data

Lat	117188	$[117188 / 100000 = 1.17188] + [\text{gps_south} = \text{true}] = 1.17188\text{S}$
Lon	3682852	$[3682852 / 100000 = 36.82852] + [\text{gps_west} = \text{false}] = 36.82852\text{E}$
Light	2	2 [range 0 <-> 100] = 2%
x_Accelerator	5	5 [range -255 <-> +255] + [acc_x_neg = false] = +5g
y_Accelerator	17	17 [range -255 <-> +255] + [acc_y_neg = true] = -17g
Temp	26	26 [range -40 <-> 125] = 26 °C
Hum	49	49 [range 0 <-> 100] = 49%RH
Sound	false	Sound [Noise / Silence] = Silence
Acc_X_Neg	false	
Acc_Y_Neg	true	
GpsStandBy	false	GpsStandBy [Active / Standby Mode] = Active
Gps_South	true	
Gps_West	false	

Bytes

3 Bytes	3 Bytes	1 Byte	1 Byte	1 Byte	1 Byte	1 Byte	1 Byte
Latitude	Longitude	Light %	Acc X axis	Acc Y axis	Temperature °C	Humidity	Status Flags

Status Flags

Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
Mems Mic Silent = 0 Noise = 1	Light Sensor < 50% = 0 >50% = 1	Acc X Axis +ve = 0 -ve = 1	Acc Y Axis +ve = 0 -ve = 1	Gps Status Active = 0 Standby = 1	Speed-Km/Hr <80 = 0 >=80 = 1	Latitude North = 0 South = 1	Longitude East = 0 West = 1